

Handling loss of signal in the ILT/RTS functions – part 2

Adee Ran, Cisco
Leon Bruckman, NVIDIA
Luz Osorio, Nokia
Jeff Slavick, Broadcom
Matt Brown, Qualcomm

Tom Huber, Nokia
Mike Dudek, Marvell
Marco Mascitto, Nokia
Mike Sluyski, Cisco
Jeffery Maki, HPE

Gary Nicholl, Cisco
Sebastien Gareau, Ciena
Xianyun Meng, HPE
Arun Kumar Alapati, HPE

Background

- Part 1 ([ran_178b_01_260414](#)) presented the problem – LOS is not addressed by the D3.0 specification.
- Straw poll #2 indicated a preference to specify some behavior.
 - However, the options presented were unclear.
- We have multiple detailed proposals
 - Will be presented briefly
 - The challenge is to decide!

Straw Poll #2 (Ran)

- When an LOS event is detected anywhere in a path that uses APSU, my preference is
 - A. Allow recovery from LOS without a full restart
 - B. Add a restart of training within the state diagram
 - C. Keep the current specifications (requiring restart by management)
 - D. Abstain

Results:

A: 5

B: 8

C: 1

D: 5

The chair observed prior to taking the straw poll that a number of the individuals during the straw poll debate time had indicated that they were confused by what was included in each of the different options. The chair asked the presenter if he still wanted to take the straw poll given the observed confusion. The presenter he indicated he did, and the straw poll was taken.

See [minutes_178b_260414_unapproved](#)

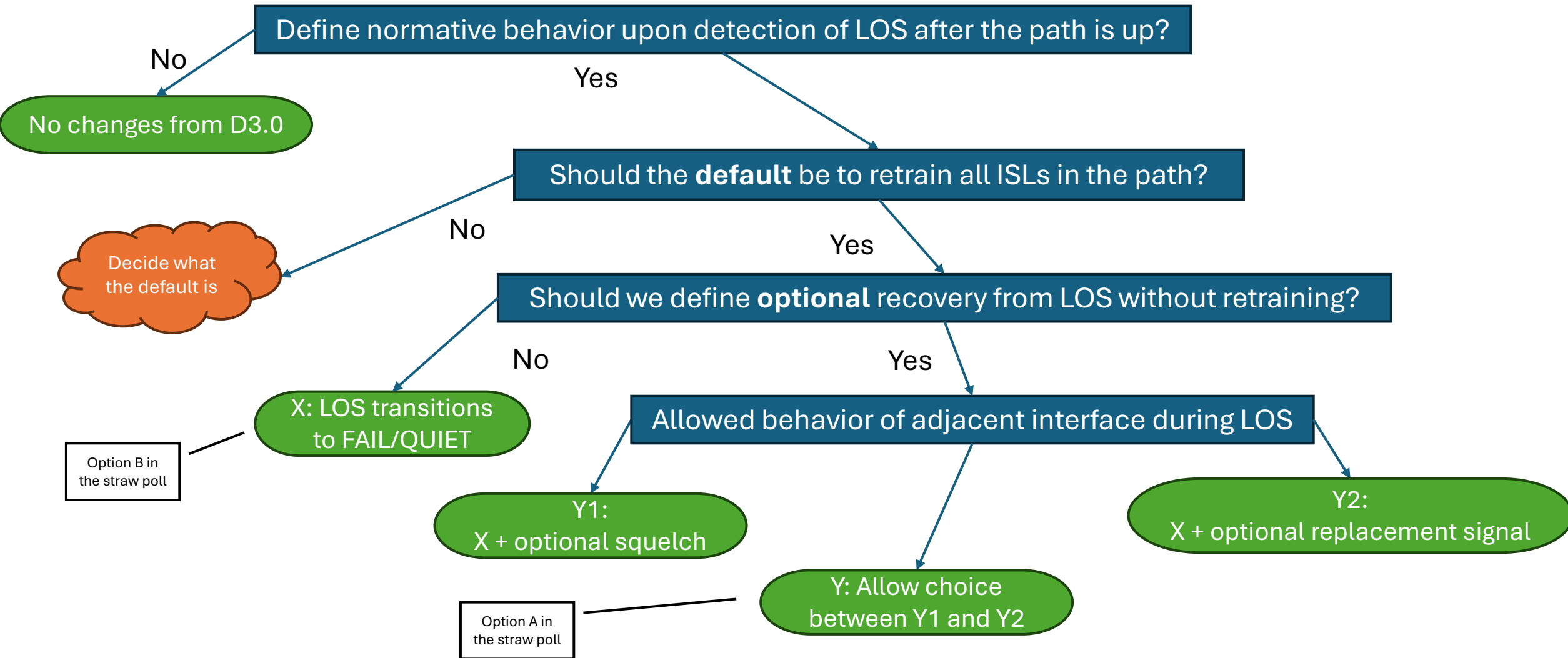
Observations

- When LOS is detected, **restarting training** is the safe way to recover
 - Provided that it causes the peer to restart too
 - If this is the only specified behavior, it requires restarting APSU on the whole path
 - **Proposed as default behavior**
- Some applications may benefit from **recovery without retraining**
 - This should be a valid option, but behavior should be specified
 - **Proposed as optional**, needs to be enabled by management
- Does management need to be involved?
 - Transition to FAIL requires management to restart
 - Alternatively, transition directly to QUIET

Goals

- Add specifications in Annex 178B to:
 - Enable a receiver in PMD/AUI to handle detection of LOS by either:
 - Restarting training (default)
 - Recovering when the signal reappears without re-training (optional, only if enabled)
 - If training is restarted, it must cause the peer interface (ISL partner) to restart training too
 - Define the behavior of the adjacent interface
- Address various application needs
 - Multiple behaviors were discussed
 - Choice may depend on implementation or application
 - Are we sure we know the right answer?

Decision tree



Proposed changes

(details in the backup section)

Split the training control state diagram into two parts

This is “part 1” (Figure 178B-10a)

Note: No technical change in the states above PATH_READY (represented by the “B” pointers)

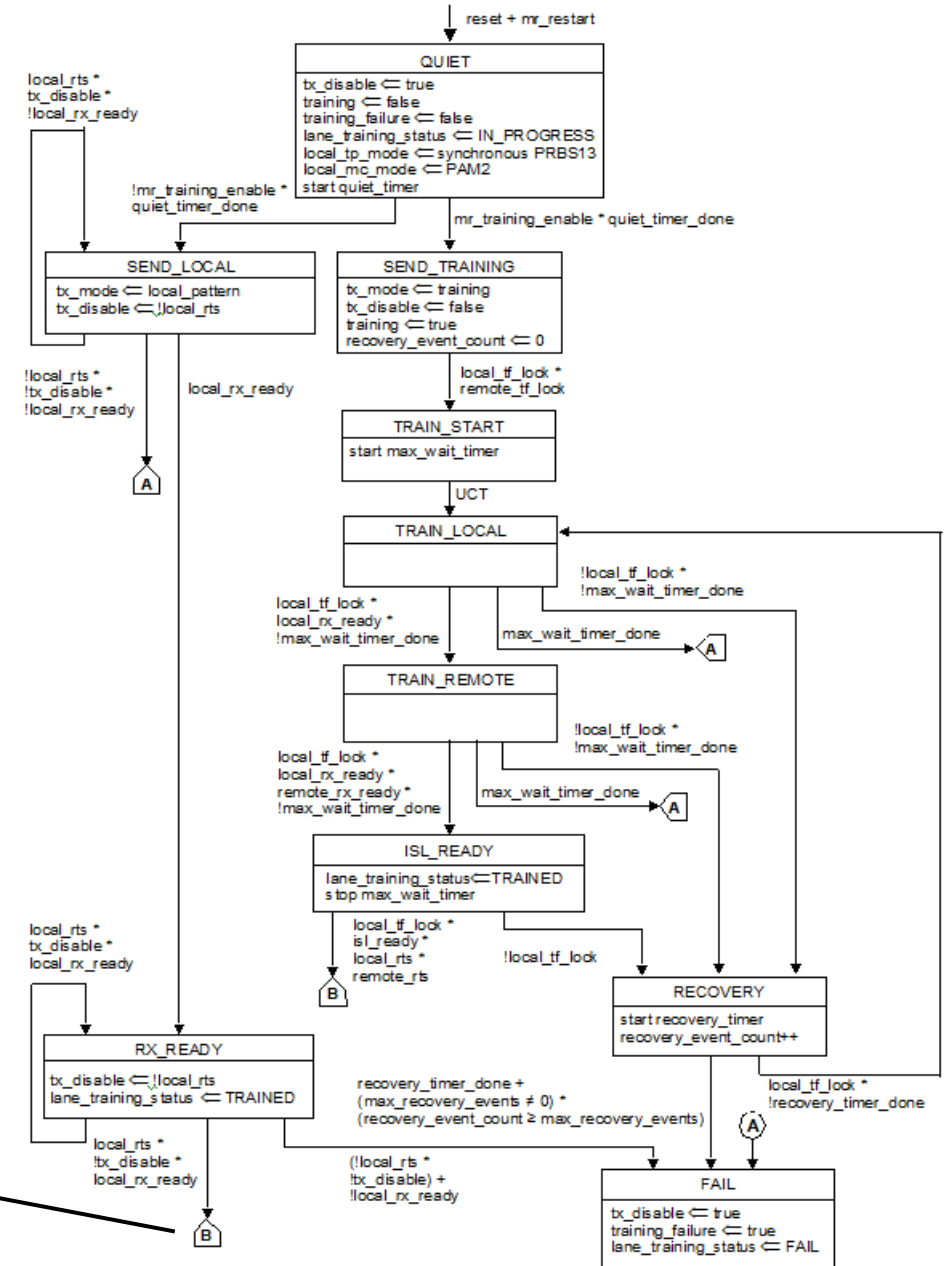
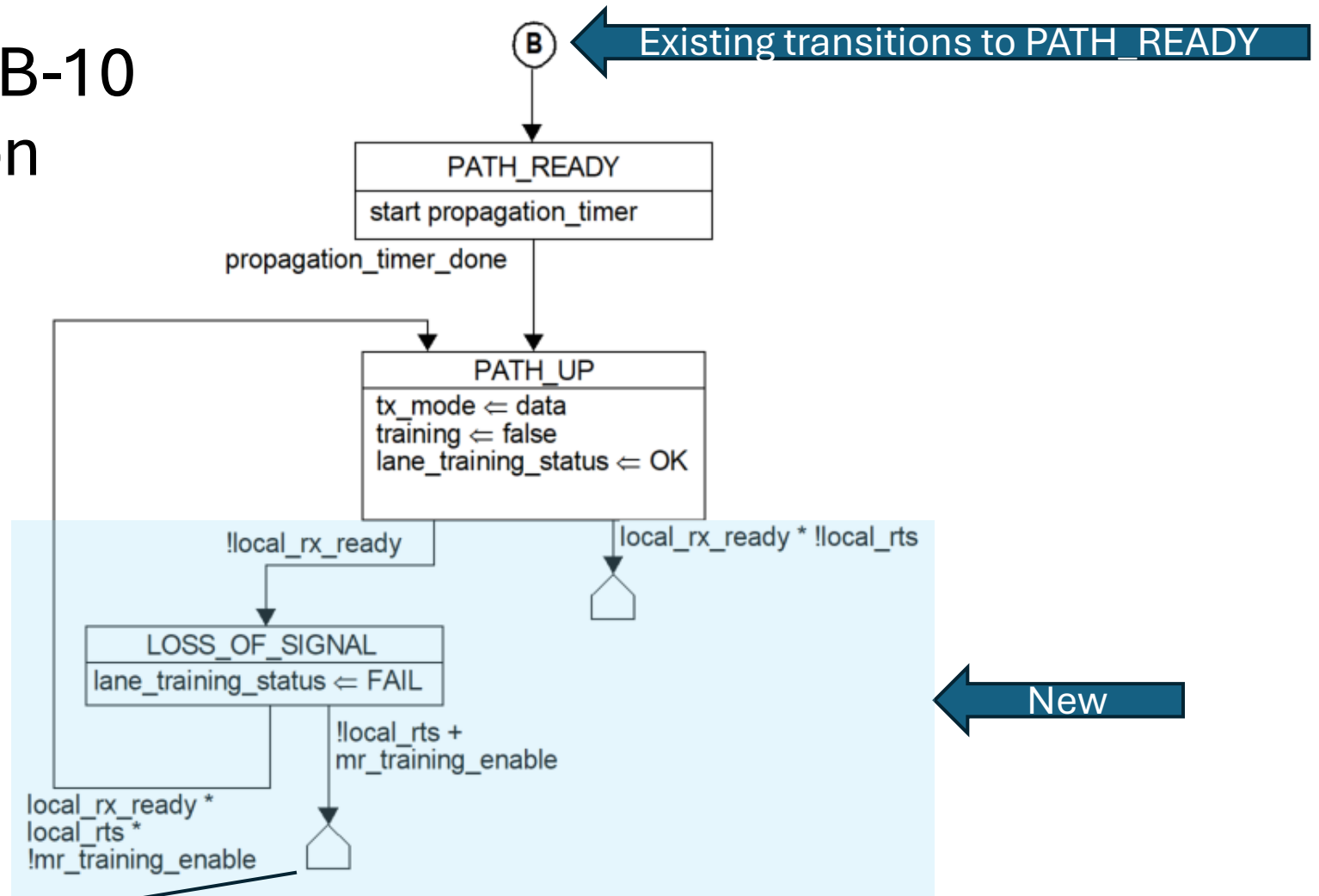


Figure 178B-10a—Training control state diagram, part 1

Additions to Figure 178B-10 to enforce a restart upon LOS

Option X



Auto restart: transitions to QUIET
 “Managed” restart: transitions to FAIL

Figure 178B-10b—Training control state diagram, part 2

Additions to Figure 178B-10 to enable recovery from LOS (using squelch or replacement signal on adjacent interface)

Option Y

Auto restart: transitions to QUIET
 “Managed” restart: transitions to FAIL

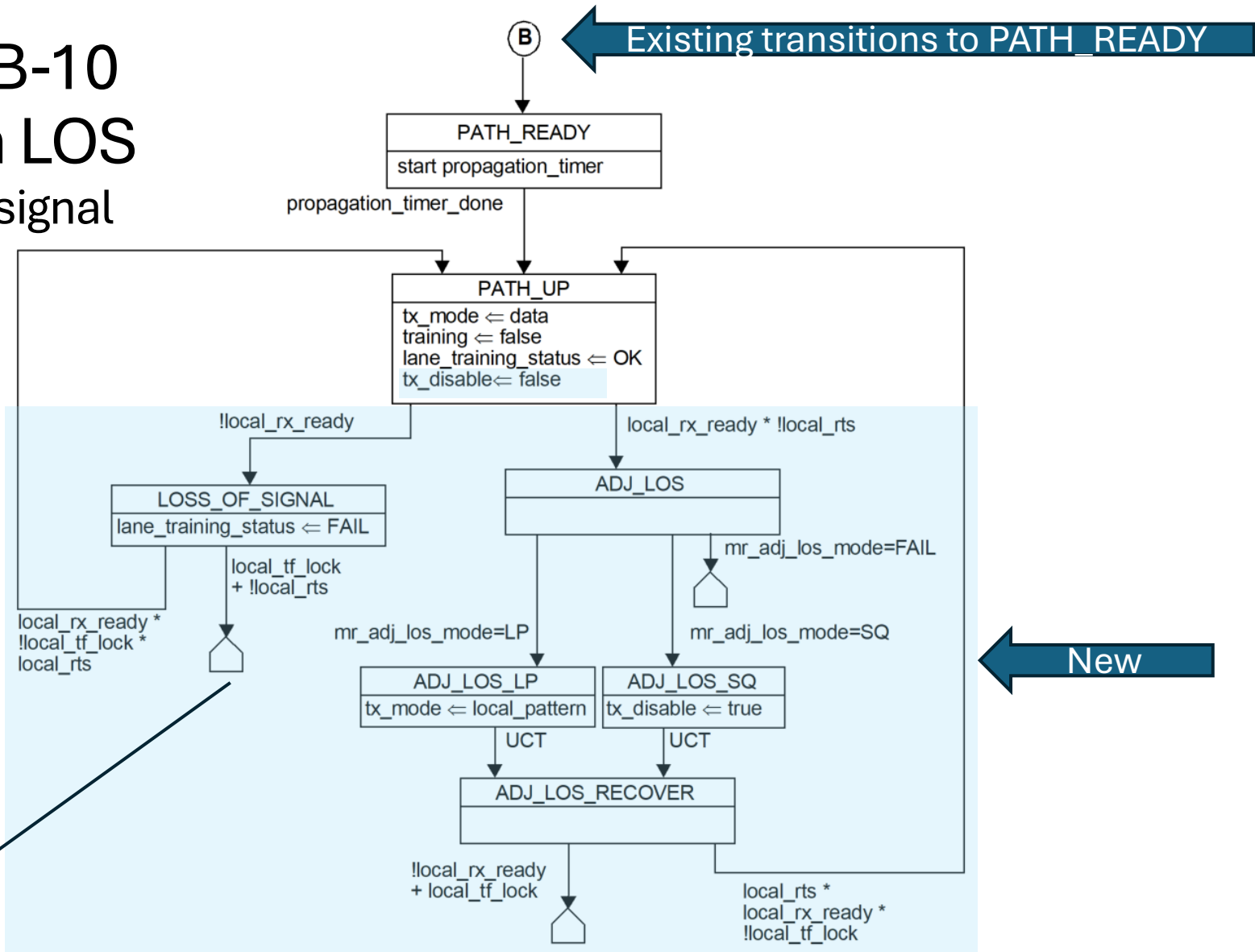


Figure 178B-10b—Training control state diagram, part 2

Notes

- With option X, restart always affects both interfaces of a retimer:
 - The one with LOS: !local_rx_ready causes transition to LOSS_OF_SIGNAL (in which assignment of lane_training_status notifies the adjacent interface) and then to FAIL/QUIET
 - The adjacent: !local_rts causes transition to FAIL/QUIET
 - The restart propagates in both directions, so the whole path is restarted.
- With option X, Remote Fault signaling is terminated in modules.
- Interoperability
 - Option Y includes option X
 - If a device that always restarts (“option X”) is connected to a peer that supports recovery (“option Y”), the peer will see the QUIET period followed by training frames – which will cause a restart
- **See backup slides for mode details.**

In this proposal we address the “unrecoverable faults”

- In 178B.8.2.1, change the note after the definition of mr_restart:
 - NOTE — There is no specified time limit for the APSU. Restarting APSU during training might result in live-lock situation, ~~thus APSU should only be restarted if there is an indication of an unrecoverable fault. The definition of unrecoverable fault is beyond the scope of this standard.~~

Required with either option X or option Y

local_rx_ready

Needs to be set to false if the peer restarts – which creates a QUIET period of at least 100 ms

- In 178B.8.3.1, change the definition of local_rx_ready :

Boolean variable that is set to true when the receiver on a lane of the interface has determined that the peer interface transmitter is transmitting a PAM4 signal, that the remote transmit has been optimized if mr_training_enable is true, the local receive equalizers have been optimized, and that no further adjustments are required for normal data transmission.

The specific conditions for setting this variable to true are implementation dependent. It shall be set to false when an input signal corresponding to tx_disable=true in the peer is received for 100 ms (the minimum duration of the quiet timer) or longer.

Required with either option X or option Y

remote_rx_ready

Needs to be set to true when not in training

- Change the definition of remote_rx_ready in 178B.8.3.1:

Boolean variable that indicates the value of local_rx_ready on a lane of the peer interface. If mr_training_enable is true and training is true, it is derived from the receiver ready bit of the status field of received training frames on the corresponding lane of the interface. Otherwise it is set to true.

Required with option Y but correct for option X too

mr_adj_los_mode

New variable

- Add a new variable definition in 178B.8.3.1:

mr_adj_los_mode

Enumerated variable that controls the behavior of the training control state diagram (Figure 178B-10) in PATH_UP state when local_rts becomes false. This variable is assigned by management and can take one of the following values: FAIL, SQ, or LP. The default value is FAIL. Implementation of the behavior corresponding to either of the values SQ and LP is optional.

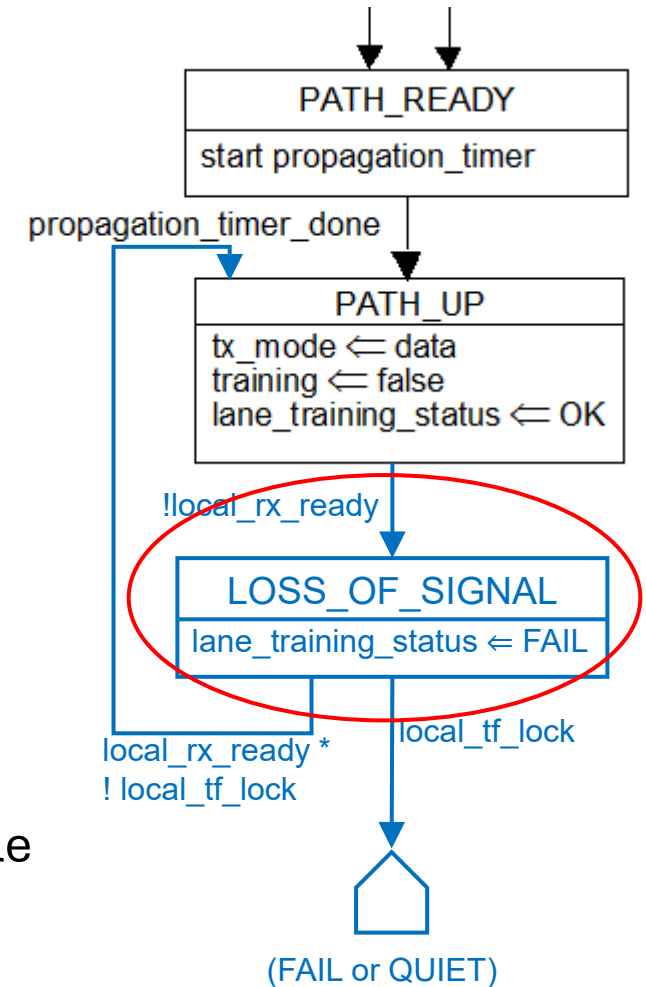
NOTE—Setting mr_adj_los_mode to SQ or LP might not be compatible with all peers.

Required only for option Y

Backup

LOS at the input of this interface

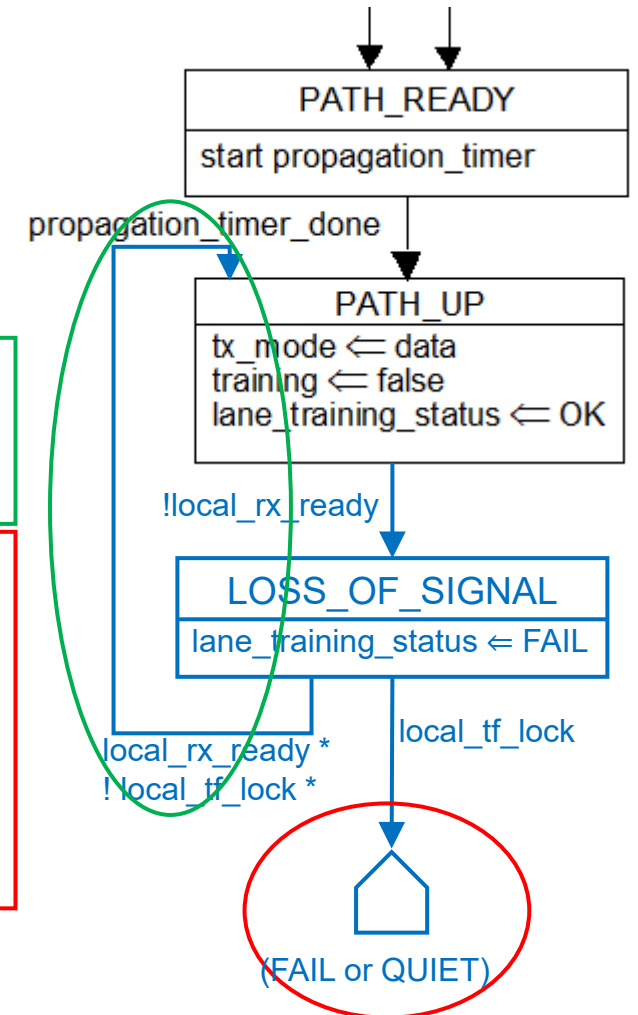
- LOS needs to be communicated to the adjacent interface to enable auto-squelch.
- In PATH_UP, when loss of signal event is detected, transition to a **new state LOSS_OF_SIGNAL**.
- In this state:
 - lane_training_status becomes FAIL
 - This is communicated to the adjacent interface via SIGNAL_OK
 - The output of this interface is not squelched (unlike in FAIL), to enable Remote Fault signaling to be used.



addition to Figure 178B-10

Recovery from LOS at the input of this interface

- Signal re-appearing is indicated by local_rx_ready
 - The timing and conditions to declare local_rx_ready is not specified. It should indicate “good data”.
- If the signal is not training frames (likely, mission data):
 - Return to PATH_UP
 - lane_training_status=OK propagates through service interface
- If it is training frames:
 - This means positively that the peer has restarted training
 - We must restart training too (no other way to recover)
 - Transition to FAIL, which requires mr_restart to exit
 - Note: indicated in the diagram as local_tf_lock, but could be detected by other means, e.g. not detecting AMs.
- Management can always restart training (mr_restart)
 - For example, if the signal does not re-appear within some time
 - Management can also decide to restart upon entry to LOSS_OF_SIGNAL



addition to Figure 178B–10

LOS at the input to the adjacent interface - problem

- Indicated by SIGNAL_OK=FAIL at the service interface (see previous slides)
- Example from 180.3 PMD service interface:

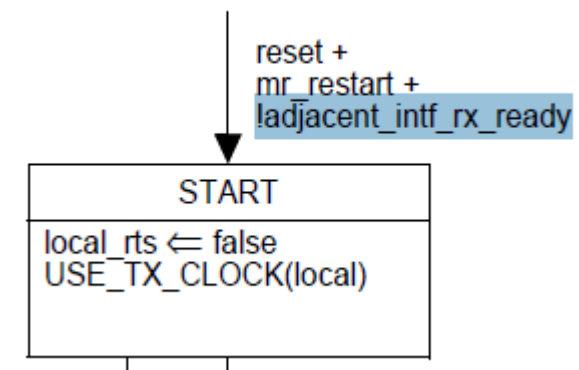
<...>

The SIGNAL_OK parameter of the PMD:IS_SIGNAL.request provides the status from ISLs above the PMD.

When SIGNAL_OK becomes FAIL, **the current interface has no valid data to transmit**

The RTS update state diagram (Figure 178B-9) goes to START, causing local_rts \leftarrow false and a switch to local clock (see below).

What should the transmitter do?



LOS at the input to the adjacent interface – solution – option X

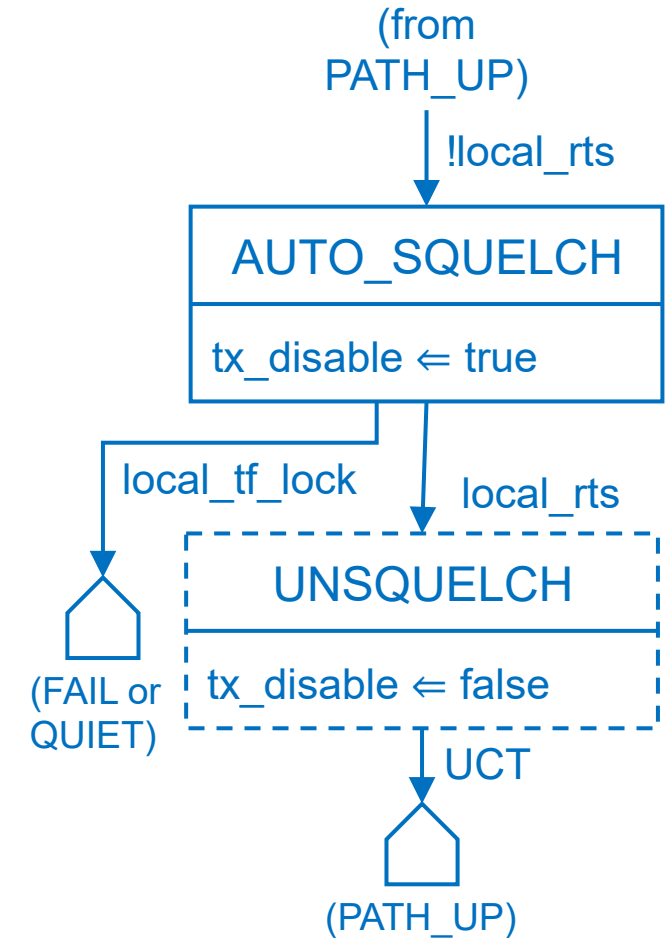
- Make restart of training (via FAIL or QUIET) the specified behavior.
 - If training is enabled, fail both local and adjacent interface.
 - If training is disabled, fail only the adjacent interface.
- If transition is to FAIL, the interface will stop transmitting and indicate the status. Management is responsible for completing the restart.
- Adds one state + one new transition to the training control state diagram.

LOS at the input to the adjacent interface – solution – option Y1 (squelch)

While in PATH_UP:

- When tx_disable=false (normal operation), if **local_rts** becomes false (indicating SIGNAL_OK on the adjacent interface is FAIL*), assert tx_disable (“squelch”) on the current interface.
- When local_rts becomes true, de-assert tx_disable (unsquelch) and return to PATH_UP.
 - Note that the UNSQUELCH state is not required if we add “tx_disable ← false” in PATH_UP.
- Detecting training frames requires restarting training (FAIL followed by management).

* local_rts is set to false by the RTS state diagram when adjacent_intf_rx_ready is false.



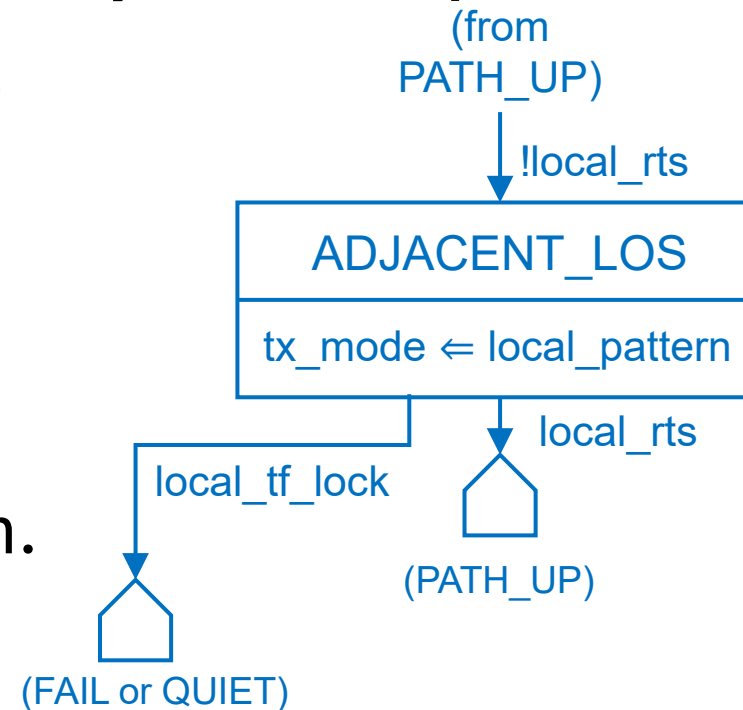
Another addition to Figure 178B–10

LOS at the input to the adjacent interface – solution – option Y2 (local pattern)

Similar to solution #1 but instead of squelching, switch to transmitting a **local pattern** when local_rts is de-asserted.

This happens after the transition to local_clock in the RTS state diagram.

When local_rts is back, return to PATH_UP (in which tx_mode is set back to data).



Another addition to Figure 178B–10

LOS at the input to the adjacent interface – solution – option Y (Y1+Y2)

- Allow either **squelch** or **local pattern**, depending on a management variable (**mr_adj_los_mode**).
 - mr_adj_los_mode is only relevant for modules/retimers.
 - It should be controlled by host management, e.g. using CMIS.
- Discussions in the consensus group showed preference to allowing both options.
- Adds 3 states to the training control state diagram.

Recovery from adjacent interface LOS

- When SIGNAL_OK on the service interface (from the adjacent interface) recovers to OK...
 - **adjacent_intf_rx_ready becomes true**
 - The RTS update state diagram advances from START
- The state diagram needs to set local_rts=true to make the local interface unsquelch ([slide 20](#)).
- If **mr_training_enable=false**, the transition is unconditional.
- But if **mr_training_enable=true**:
 - If a recovered clock is used (module/retimer) it needs **isl_ready (local_rx_ready * remote_rx_ready on all lanes)** to advance to SWITCH_CLOCK
 - In data mode, training frames are not received, so **remote_rx_ready** is irrelevant. It should be set to true in its definition.

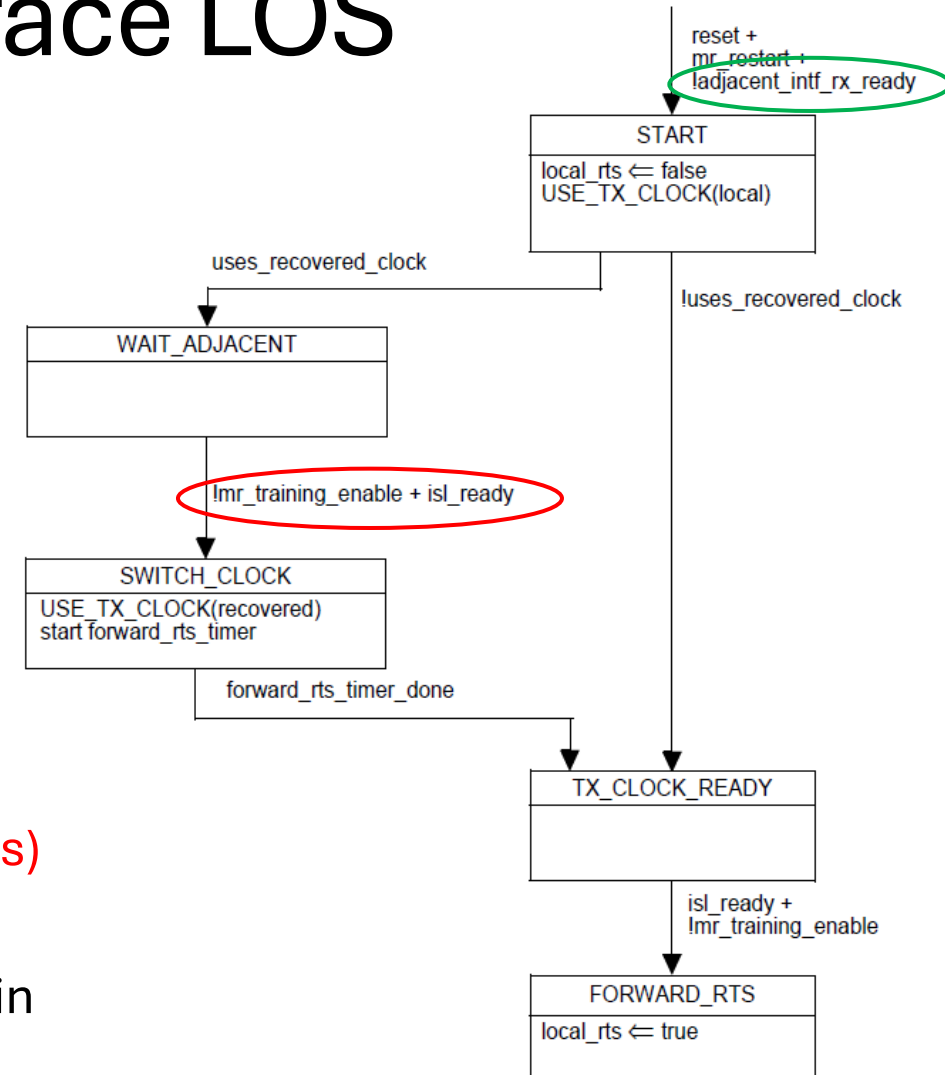
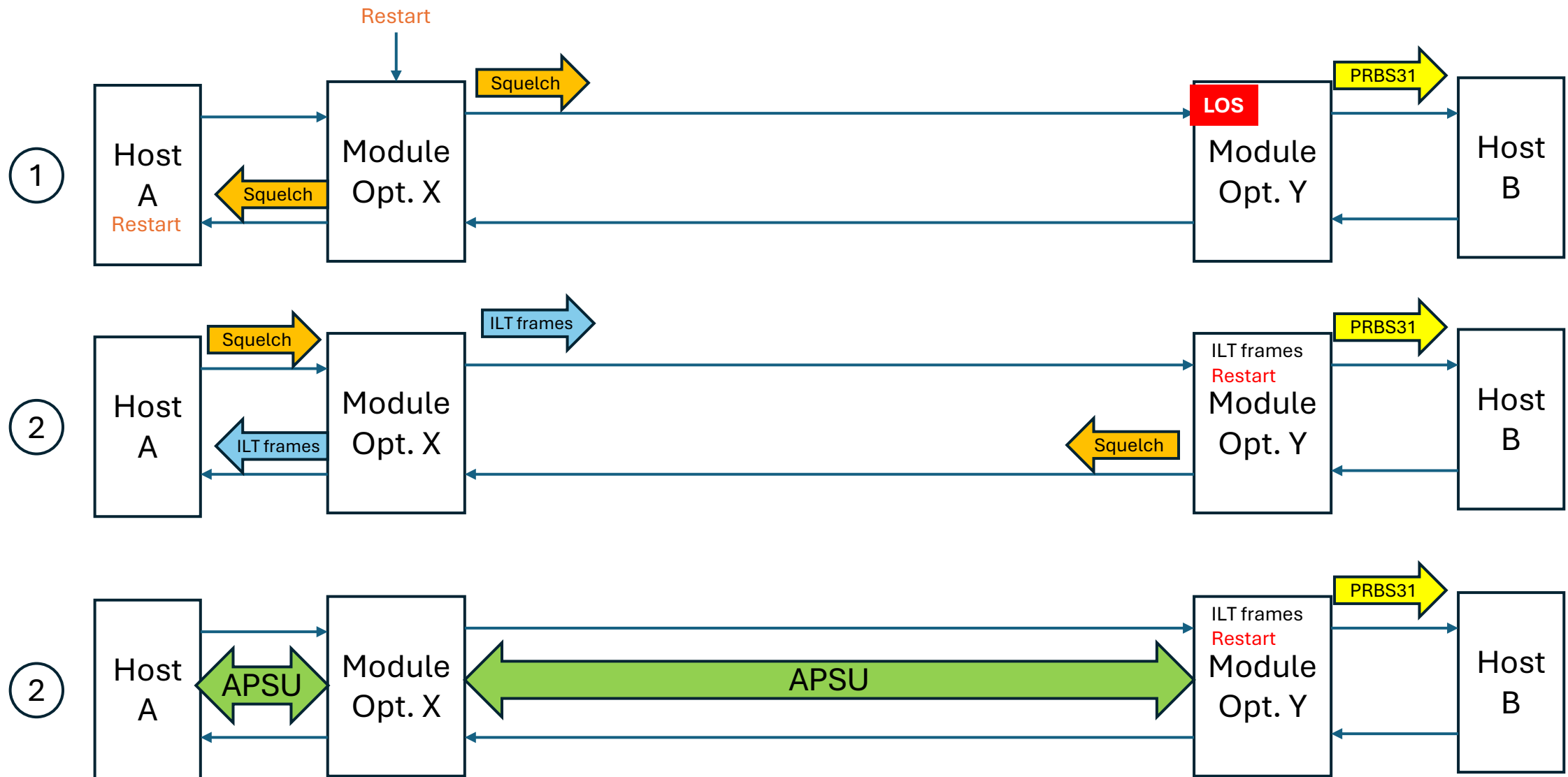
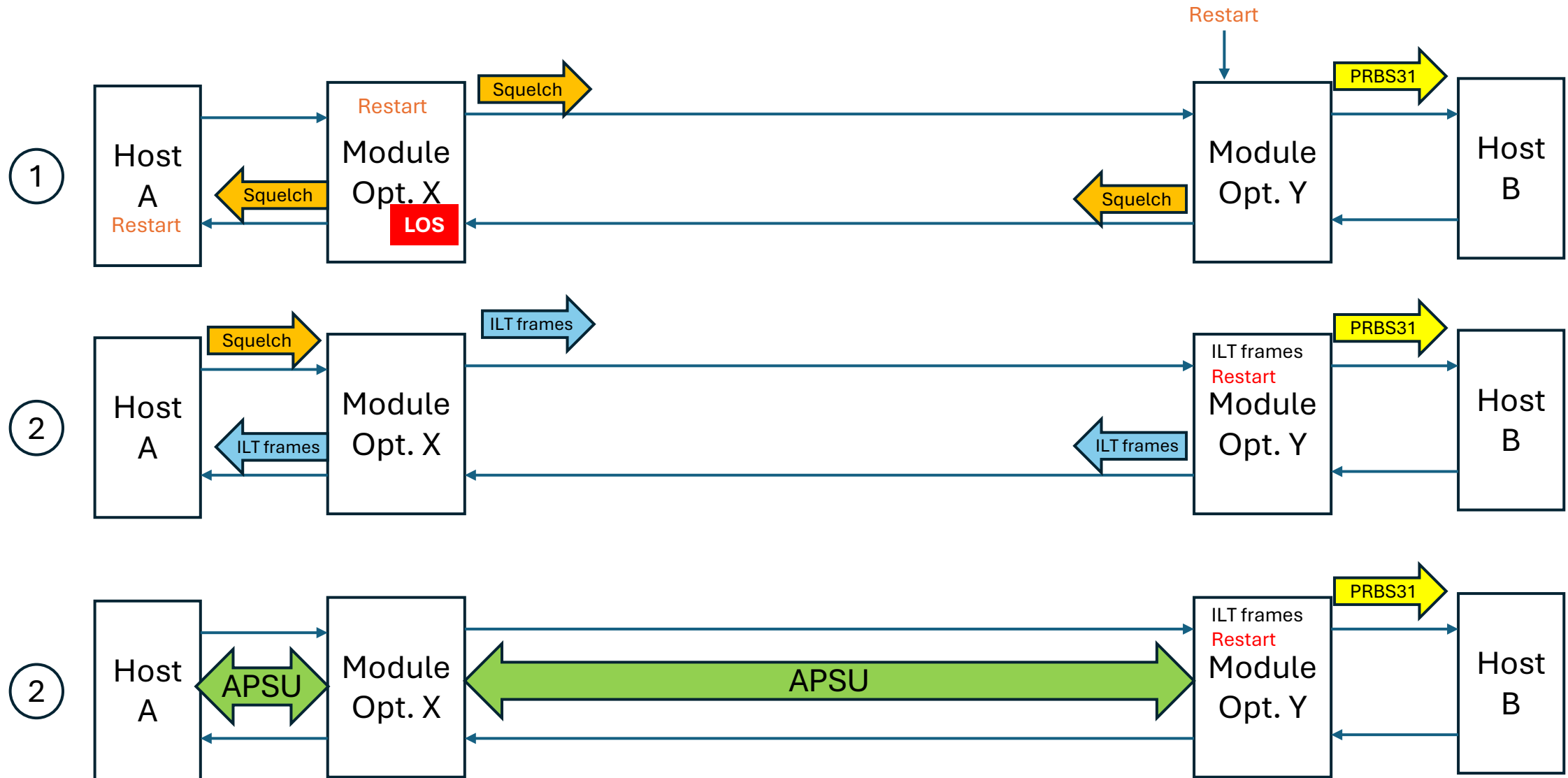


Figure 178B-9—RTS update state diagram

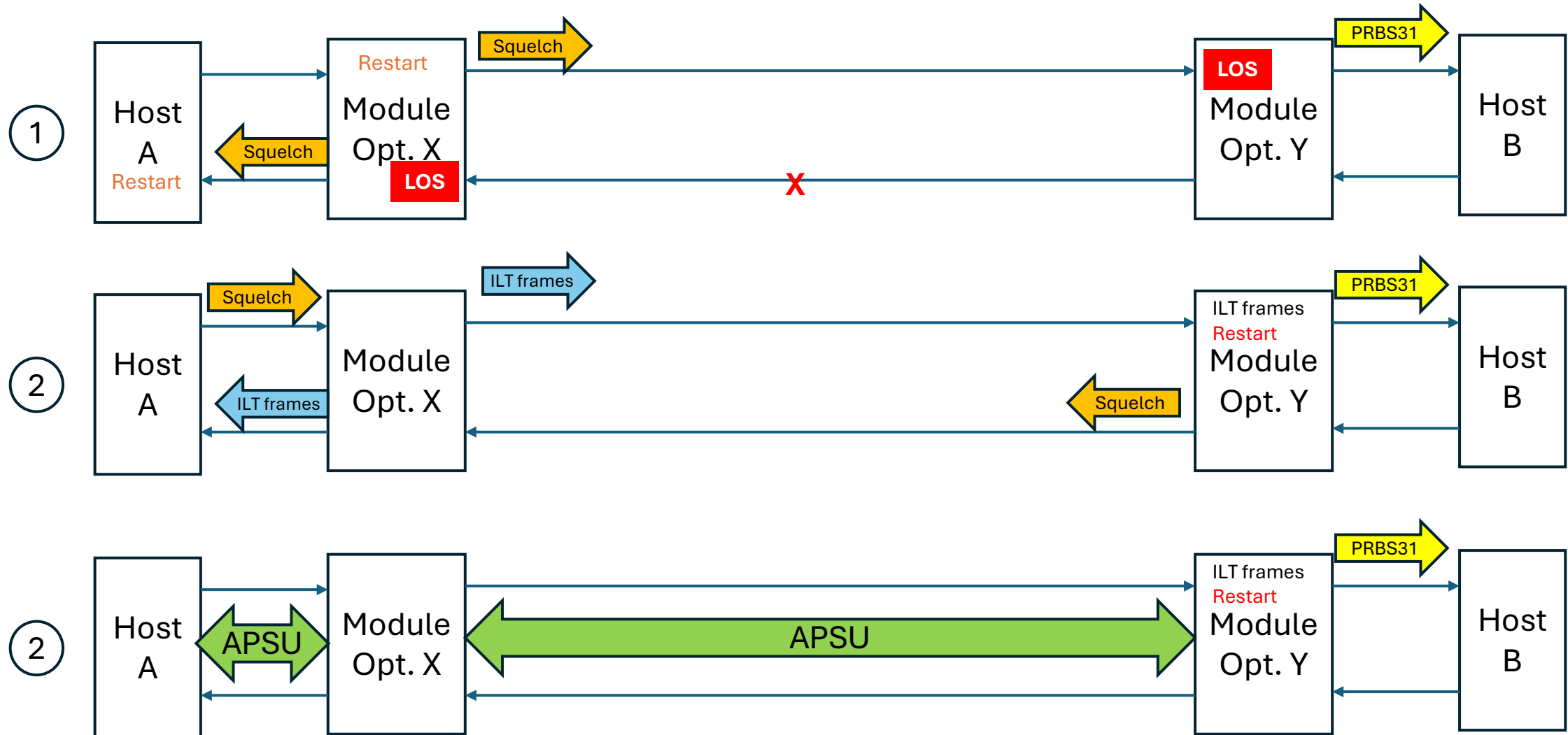
Option X – Option Y interoperability – Case A



Option X – Option Y interoperability – Case B



Option X – Option Y interoperability – Case C



Option X – Option Y interoperability – Case D

